VISITS Lessons Learned Wrap-up Session
June 1, 2006

In attendance: Anna Marie Moe (MPD), Brian Pierce (MPD), Thurston Elfstrom (MPD), Corrie Hahn (MPD), Dan Bethke (MPD), Derek Scoble (MPD contract PM), Jamee Fields (ITSD), Deb Knudsen (scribe)

These are the notes from the VISITS lessons learned exercise. VISITS is a project that kicked off in the January 2005 timeframe; it basically took almost 16 months to accomplish compared to the proposed 5 month implementation period.

After we started talking about all the associated lessons that we learned we decided that recording it would be a better method to capture the information (verses a template) to allow freedom of expression with indications of performance Weak to Strong. The template focus areas of the Lessons Learned template are identified by **Bold** underlined text following the summary comments.

SUMMARY COMMENTS

WHAT COULD WE HAVE DONE BETTER?

1. **Did not have training for these project activities, had to learn by doing.**
2. **Requirements evolved from soft to hard.** Having four miss starts before the awarding the proposal, afforded us quite a bit of learning room to solidify the requirements.
3. **Learned how to make strong requirements.** We still had room to make the requirements harder. One of the things was to make the requirements more specific. We didn't realized the specificity to which we needed to take it; the level of minutia. An example that we talked about was during one presentation by the vendor when we were getting the first look to see how the software would function. At that point we were observing on how many steps that it took to go in and insert a contact record, search against the database and do an Invite-a-Friend. We were struck by how lengthy and time involved that was. There would have been an opportunity to give a specific increment that this process should not have taken more than 5 steps; specified how many screens could be allowed in a step. So could have controlled the process and gotten them tighter within our RFP. Since we didn't, it allowed the vendor the opportunity to take a different interpretation. On paper the vendor could justify saying the to say that they had met the requirement even though the replacement activity which only took 4-5 steps in the existing system now took 30 plus steps in the replacement system. We didn't have a good legal perspective to stand on in that particular aspect. We speak the same language; we understand how our business operations are. But when you put those same requirements into a textual format, and we expect vendors to understand our interpretation, that is not always the case.
4. **Proper requirements engineering (development – test) is necessary for a project success.** There is no good way, in the approach we took, to make sure

that everyone had the same interpretation even though we went to great lengths to try to explain requirements. We even put the requirements into the RPF as an attachment and told the perspective vendors to tell us how much level of effort is required in order to customize your existing COTS package to meet the requirements.

5. **Spend the proper amount of time for technical environment engineering.**
6. **Be very definitive about design, better understanding – about screens, action steps.**
7. **Understand the CRM space better.** Maybe we didn't understand the CRM space well when we first started. Maybe we should have hired a CRM expert to have jointly defined requirements in a way that would have better served us going forward.
8. **Fulfillment – The whole inventory was not tested. The time it takes now to do labels compared to before is excessive.** We weren't able to test a lot of the functionality because we didn't have a separate high-speed dot matrix test printer to use to print reports and half of the reports weren't done when we went into production. The whole fulfillment/inventory side was not tested very well. That was one of those areas that the steps involved were so much greater than what our expectations were. It is a terribly protracted task. That is another area where we could have tightened requirements by potentially putting a time box around it based upon the current process. In order print bulk mailing labels on average it takes X amount of time. It takes 4 hours to do currently 300 labels what it took 15 minutes in the old system to run consumer labels. The fulfillment side was the most customized. Initially it was suppose to be vanilla, but they did so much customization to the fulfillment side that that was the last piece that they got up into production and the piece that we had the least amount of time testing even when we were in the testing phase.
9. **We only had one high-speed dot matrix printer, and it was in use.** So we didn't have the proper test printer. The problem is that we only had one printer; it was cost prohibitive to buy a printer just for testing. The original requirement in the contract was to have them write the application to talk to our existing printer. That never happened. That is part of the problem with the change request #3 because it can't talk to the printer in the way they want it to.
10. **Performance – Invite a Friend, Web Chat, slowness.** We thought we did everything right as far as setting this project up for success from a contract perspective and from a requirements perspective. In one respect we were saved in the approach that we took with testing only ½ day before we called it and said that the system wasn't ready for system acceptance testing on the original schedule. MPD stuck to our guns. It was a very protracted process. When we talk about the Invite-a-friend number of steps, fulfillment, web chat, slowness and complexity of the solution set to what we originally had in mind, MPD now has a great lumbering, very powerful application that is not nearly as nimble or responsive as what we originally intended.
11. **The vendor gave us a requirements doc, they gave us the minimum system requirements and not the specs we needed.** One of the problems is that we asked the vendor to provide us with the specs that we needed for the hardware

that we needed for this. What they gave us was the requirements document. It was the minimum requirements versus what we thought were the recommended requirements. We bought as close to the requirements as we could for cost reasons thinking that they were the recommended requirements. It was not adequate; it was the very minimum usage requirements. It doesn't function properly. Our environment as purchased now may be throttling down the ultimate performance of the system we had originally anticipated and the cost associated with stepping to that higher level is fairly expensive. It is not a cheap solution. At this point it would require buying all new servers for it. When if we had know about it earlier we could have spent an extra $10,000 at the beginning and gotten better machines. During the project we have already added more hard drives and RAM to increase performance. We're not convinced that it would even help step to the level that we all had from an expectations point. Although, it wouldn't have solved the problem, it would have helped.

12. **MPD asked for hardware requirements in advance but we didn't have the full hardware requirements till they asked for it when they wanted it.** We never got a whole report for all the hardware required when we asked for the hardware required. In the middle of the project we were always buying new equipment that was never said we were going to need from the telecommunications subcontract vendor standpoint. From the Siebel side they did give us just the minimum requirements. We asked for the requirements for the whole solution so it should have included both the telecommunications subcontract vendor and Siebel components. There was hardware that wasn't even mentioned until it was referenced in an email that was received on Friday that a Linux box was suppose to be installed before Monday. There was no opportunity to order anything.
13. **Asked for 3rd party software requirements in advance but received those late also.** We were lucky that we were able to buy software on line. We ended up buying 2 copies of Linux because they didn't tell us which version to buy. The first version was too new to be compatible, so we had to buy an older version.
14. **Any request for remote configuration, reliance on subcontractors.** The biggest mistake we discovered is to allow any request to do remote configuration. Remote configuration removes documentation. We lost all of that configuration documentation. That was a big problem. The project as constructed by the vendor created reliance by the prime on subcontractors to help them implement the solution. We lost a lot from the perspective of having a coherent firm hand on execution when the prime said they were responsible for managing all the subs in here that are bringing this functionality to support us at the call center. MPD took the position of it was a problem that the prime needed for find a solution for. The warning signs were there. We were working side by side and knew that what we were doing was working so it created a false sense that everything was working.
15. **Vendors job to get things configured.** The vendor tried to transfer risk back to MPD later when things weren't going well.
16. **No warning signs, from the implementing vendor "We can't actually show you," We should have stopped.** That meeting that we got the first look at the system, other than Corrie, we naively gave the vendor a lot of rope. They are

saying that this is just the first look, and we believed them. They would say that this is the vanilla screen view, but it will be changed to look like what you said. During the design meetings before signoff, when the vendor first said we can't actually show you that now; we should have said "end the meeting; come back when you're ready". That would have put the ball back in their court and made them fully responsible for design decisions.

Even though the vendor promoted team responsibility, they had their meetings behind closed door. Nothing was ever said to the customer that things were going wrong in advance of UAT.

17. **The vendor was not open and honest; tell us what you are doing or need.** If they would have been open and honest that things were going wrong since we were not on a time crunch, we could have managed the relationship expectations better. We stepped forward many times to fix their stuff. We would have stepped forward again to help them do what ever we could, but they broke trust.
18. **The Vendor constrained themselves, couldn't pull this off. Didn't share problems.** The vendor realized that this was not going to be profitable project and started to manage the bleeding not the quality. The first evidence we had was when they don't start share as much as we thought that they should. So we were blindsided.
19. **Video tape meetings, visible evidence; we gave them benefit of the doubt.** In retrospect we should have video taped the design meetings. There were verbal promises made during those exercises that evaporated over time. We would have had the visible evidence that we raised issues that the Siebel design was 3 times more cumbersome than the legacy system; they made promises that the system would function better. We would have had visible evidence associated with those promises. MPD gave them the benefit of the doubt with good intentions because we were trying to work as a team and as a team we needed to allow for slight adjustments to be made especially in IT projects—where it is not a cookie cutter approach. MPD was a good partner. Vendor reference checks were excellent. No indications of any of this in advance. They were a top flight vendor. They are one of Siebel's top partners.
20. **The technical team was very good.** Except for the sales team that originally responded to the RFP that didn't understand the requirements, the technical team that stayed here was very good.
21. **We had the right to accept technical team members.** In the contract we had the right to approve or disapprove anyone coming on or off the project and subcontractors coming on or off the project. We gave them the benefit of the doubt. They had 3 Project managers; scripting—that lost some stuff on the initial data load; people floated through here because they were on the bench and the vendor was rotating them through. We knew that they were behind, and they wanted to bring more resources in. They were saying the right thing, unfortunately we experienced a couple of areas of the vendor execution that came back to bite us. When they realized that they were starting to lose it, we saw them staff up and bring in extra resources. Some of them weren't proven, and we paid a price. When we lost stuff on a load the Vendor owned up. One was with the resource that was improperly applied--someone that was not ready to do that

work; the other was the fact that the vendor got into a bad situation with one of their subcontractors and at some point they weren't really talking to each other which was really keeping the entire project from moving forward on some key functionality that we were interested in.
There weren't supposed to be any subcontractors, but we allowed it since they didn't have the expertise in one area. They pitched it as part of their RFP proposal. The problem was that after 4 years to get to this point, we weren't in the mood to make any changes when we were about to ink somebody that on paper was a top flight implementer.

22. **RFP Price cap, Reached out to the smaller vendors.** With the price cap, MPD thought it would get rid of some of the behemoth vendors. MPD tried to reach out to the smaller vendors. The actual software is not the biggest cost in the project. They were able to use Siebel and get it under the cost cap. Siebel probably wasn't the best fit for the project since it is not made for such a small group. In large organizations there is more division of labor.
23. **Price break point set too low.** The price cap wasn't a bad idea, but we were on a bubble with the price cap. We either needed to lower it or raise it. It allowed Siebel to come in with an incomplete implementation.
We may have been too tight on cost and it was one of things that probably bit us in the end. These guys sharpened their pencil to get to the price point that we were willing to accept. Double edged sword. If we were not willing to let them go over by a certain percentage point, we would have eliminated Siebel from the running all together. We might have caused some of that scenario to develop because we set that bar so low. When people responded, we were feeling good because we weren't going to be over spending. But in reality we were forcing people into the situation where they couldn't deliver when they had told us that they could. We found ourselves in a rough spot. How to believe vendors when they say they can deliver for a price?
24. **Siebel many path-ways –** Siebel is a behemoth. There are so many different pathways to implement. Maybe the initial design team didn't understand it well enough to design it right. In our mind how could a top rated CRM be so cumbersome to run labels; it is so ridiculous we never thought that would be a problem. All we heard was that Siebel was great and the top rated CRM by Gartner. As result of our due diligence and research, it only made logical sense to go in that direction especially given the space and role that we fill in the state of Montana and we wanted a customer focused application.

    Part of the problem is the amount of scripting and customizing and the strategy the Vendor took. They would come in and say that they had a couple of different ways that they could do things and we let them do what they wanted because they were the experts. In hindsight we probably should have asked the ramifications of taking option A versus option B were. What is that going to prohibit us from doing later on? We could have been more sensitive and done a better job specifying complexity and performance parameters. If from an implementer perspective if you say you have to do it a certain way that is fine as long as you meet the performance requirement, as long as you meet the complexity

requirement. It should take no more than 5 steps and 10 seconds to kick off a job. If we would have done that, they could not have built the system the way that they did because it would not have passed the performance requirements. You are hiring an implementer that is supposed to be bringing their best solutions to you. We don't know if it was the best solution, we know for sure it is not from a performance or ease of use perspective to best solution. Even subsequent vendor people have asked us why we did things a certain way because then we couldn't take advantage of other functionality. Our only response was because your people designed it that way. We thought we did the right thing, we found a good implementer with a great track record who said all the right things. It is the difference of opening up a package of instant pancakes within 30 seconds or making pancakes from scratch. The end result are pancakes, but have to wait a longer time to get and more cumbersome. Without having that expertise in house how would we know? We had to trust. We needed more emphasis on system performance and complexity on the front end requirements. We don't care how they do it as long as they do so it performs.

25. **Emphasis on performance, complexity, time.** We know our business; we could have set our performance and complexity rules. We know we can do this job in this many steps and it takes this amount of time. We really don't care how it gets done as long as it gets done efficiently. We wouldn't have had to have that knowledge. We did a recording of calls in Missoula and wanted to use it as a test against the new system. If we had those types of specs in the RFP, how long it takes to process a call; how long does it take to do an Invite-a-Friend, and we had a record of what that was in the existing system we would have really helped ourselves. We would have found much earlier potentially that Siebel as implemented wasn't going to meet the goals. We tried to do everything right with open relationships, then use wishful thinking that it would come together. The system does the job, but doesn't meet performance expectations and is clunky in some places. It doesn't mean that it won't be fixed over time because we've taken on the maintenance role and will have opportunity over the life cycle to fix things before we replace the system. The call center is taking notes on paper because the system is not fast enough to capture information; and using the waiting time between calls to enter data in which was not our intent or expectation. We spent a lot of time and effort to trying to be smart. Not sure how overcome some of it.
26. **Vendor failed to provide expertise.** The Vendor failed to provide expertise that was required and asked for with in the RFP itself. There was a level of interaction that we expected to provide; there was a lot of work that they said that they couldn't do that was clearly defined in the RFP that we ended up doing for them. MPD went to the Nth degree of support to make trips over to our call center and buying equipment that wasn't originally specified. Creating email after email on web chat; spending political capital with ITSD with security issues. MPD took on one of the major components, the integration of the Tourism product database. We just did it ourselves with very little help from the vendor since it was a great way to do the integration, and it is one of the parts that works the best. MPD did 90% or more.

27. **Vendors failed to push jobs to subcontractors.** Vendors failed in pushing along project requirements to their subcontractors. Early on in the process we suggested that needed someone else to manage the process in Missoula. We shouldered a lot of responsibility for them. An example is the whole telecommunications system, in the beginning Brian was doing all of the maintenance and support for a system that he had never touched before for a system that they were supposed to be experts in. And then Dan got involved and again he had had no experience, and the vendor should have been supporting the system. Where the vendor should have been supporting the system, the vendor said instead well you are getting to know that system so why don't you go ahead and maintain it for us. They did it in such a way that you will have to maintain this so why don't you go ahead and take care of that. There is an element of truth to that statement.
28. **Personnel.** The personnel aspects of projects kicked as the vendor's project manager's mother died and he was the estate executor; the vendor's lead engineer came down with an African virus and was down for 3 months; MPD Anna Marie had some serious personal issues. We were very good about having the vendor's resources gone. Yet when we had ours gone, all the sudden we were the ones holding up the project. We dealt with them on a higher level, pointed out that it was entirely equitable given the way that MPD had accepted Rick's departure. Although that argument went away, it never should have had to have been made.
29. **Quality Control.** The vendor's main problem with MPD was with the detail that Anna Marie held them accountable with the documentation and contractual obligations and payments and things of that nature. While the guys were slugging it out on a technical level, somebody was really keeping their feet to the fire from a documentation perspective and they didn't like that. They were actually being held accountable for high quality deliverables. When a vendor realizes that they are no longer in a profitable situation, they need to start cutting corners and that ended up happening and Anna Marie held their feet to the fire. Another vendor would have fussed a lot more. This one squeaked a little, but we got there after a couple of evolutions. The vendor stood up more with the challenges than some of the other vendors that we potentially would have worked with. From a reputation point of view they deserve some credit for being standup and not backing away from their ethical responsibilities. There are a lot of other areas that we can say we were under impressed. We told most members of the technical team that from a reference point of view would give a very good reference. A lot of vendors would have cut and run. Basically since May they are working for free. When consider it cost them about $30,000 a week to be here and we estimate that they lost ½ a million dollars on this project. A lot of less reputable vendors would have cut and run. Not pretty, but they stayed. The system complexity that was not anticipated. Some huge complexity issues that was solved by this team. MPD solved a lot more than we originally signed up for. It had impacts within our own team and tested our patience.
30. **Learning.** What are the things that you learned about as staff and individuals in this project? We worked very well as a team. When we saw a problem we would get together, discuss it, assign tasks and timelines and get it taken care of. If it was our responsibility we got it done. It didn't matter how long it took; if we took

on a task we made sure that we followed through with it. We were not going to slow this project down—it was our goal. We drilled that into each other. Do not give the vendor the chance to point to us to be the problem anywhere on the project. We maintained the upper hand throughout the project because we never gave them the opportunity to pin the blame on us. One of the keys to maintaining the vendor buy in was because we never gave them the opportunity to step away; we never beat them. We never put our foot on their neck even though we could have. Everyone stepped up. Never had anyone refuse to do their part or point the finger when things didn't work. MPD stepped up to the point of letting our other jobs go to the wayside to make sure there was no opportunity to point at us.

31. **Red flag on price differences from vendors.** The system is really a $2 million project not a $1 million project. Especially if you consider all the other proposals that bid Siebel were much higher well over $2 million. It was the highest price project on every single RPF except this particular proposal. Another Vendor came in at $3 million. That should have been a red flag there. They sold us a Ferrari with the software, but they sharpened their pencils with the implementation. They really didn't have the resources. We helped create some of this by thinking we were so smart by trying to cut this way down. We were being driven by not necessarily MPD, some outside influences—need to recognize that-- create environment where we step into these problems. We then have to scramble to be successful.
32. **Look at timeline more closely** Wasn't just the money. Every other vendor that proposed Siebel, time wise, another Vendors' initial Siebel proposal was 9 months for phase 1 and an additional 5-7 months for phase 2. The chosen vendor was proposing an 18 week implementation for Siebel. That was the first red flag. If everyone else is saying 1 year to 1 ½ years to implement, how do they expect to do it in 18 weeks?
33. **"Gut feelings"** The reality is when we initially accepted their proposal there was this feeling that it wasn't realistic. It's a double edged sword on the selection team when we were looking at a timeline, with a COTS package and low customization, it shouldn't take that long to implement. This is exactly what we were looking for. Timeline, vanilla implementation and low cost should go hand in hand.
34. **Time crunch starting with a season.** One thought was that it would be ready in time for the summer season. While it wasn't the primary concern it was a thought.

WHAT DID WE DO REALLY WELL:

1. **Worked very well as a team, assign tasks, followed through**
2. **We did not slow this project down**
3. **We never beat the Vendor**
4. **Never provided the Vendor the opportunity to leave**
5. **Everybody stepped up, no refusals, no guilt about our execution**
6. **We stepped up to their problems and arranged our jobs to help out**
7. **Acquired the needed knowledge**

8. **Tourism Database and Web integrations**. One of the big goals from the inception was to fix a process in Destiny that had been broken for years which was the integration of the tourism product database. It was done elegantly and did it in a way that made it easy for the travel counselors as the end users to use. It is one of the truly shining things within the system. Getting the database letters back and to track people's interest again which is helpful for marketing.
9. **Our requirements were good**. We did have four chances to refine them, but it saved us many times.
10. **Have an outside PM and IV&V.** They bring a neutrality and experience to the table.
11. **Have contingency fund/holdbacks.** As much as you think that you've got everything in the RFP, this is a must. It should be mandatory. It was strongly suggested, but not mandatory.

WHAT STANDARDS OR PROCESSES NEED TO BE MODIFIED AND HOW:

1. ITSD if they offer assistance, they need to have someone who will assist. If they want someone on the committee they need to make sure that they have someone who wants to be on the committee. It did force us into a bad situation in the first iteration, but we received support in later iterations.
2. If ITSD has requirements that we have to meet, then they need to be able provide assistance to what those requirements need to be instead of just saying this is what it is. The rock methodology. If you want a round 3 inch rock for skipping tell us that instead of what you did is nice, but that is not what we want. They changed the rules multiple times during the project. They let us go through the process and then they changed the rules at the end which wasn't fair. They should have let us finish the process with the rules that we started with.
3. The security office needs to be privy to all these types of RFPs at the beginning when we are trying to get the RFP approved to send out. Should not come in after the fact and saying that we can't do it because it is a security risk after ITSD had already approved it and hundreds of thousands of dollars already invested. If the security department wants to have a say they need to review our RFP when the ITSD Business Analysts do.
4. The idea to have site visits is a good one, especially with big projects like ours. It should be a requirement according to costs. If projects are over $500,000 for example, it should be required to visit two sites at a minimum. Although each project is so unique that you won't ever find an exact match, it would give you an idea what they have done before.
5. Having an outside PM and IV&V should be required. They bring a neutrality and experience to the table.
6. Require contingency fund/holdbacks. As much as you think that you've got everything in the RFP, this is a must. It should be mandatory. It was strongly suggested, but not mandatory.

LESSONS LEARNED TEMPLATE

INCEPTION/FEASIBILITY

Concise, written proposal with clear statement of work and deliverables. Strong performance by MPD.

We got there, but we recognize that we could have done even better. We talked about some of the areas where we could have done better. We learned about the importance of good requirements, contracts, holdbacks, and best practices (PM and IV&V) by going through the process (longer than anticipated) of developing a good RFP.

Proposal shared with team. Strong performance by MPD.

No doubt about that since the proposal and objectives took 4 attempts before the actual award.

Development part of the proposal process. Adequate performance by MPD.

We thought we had the RFP very tight by the 4th time. However in retrospect adequate. Had we had the 20/20 hindsight vision, we would have seen that we would have included the time performance restrictions and more specificity within what we defined as hard and fast functional stipulations. We would have fleshed those out more so the vendor would have had a greater understanding of our performance requirements and complexity.

What would we change in the selection process? What would we have done from a due diligence perspective to ensure that we were picking the right developer-- the right integrator? We should have tried to find the most similar project fit in the Vendor's resume and gone and done a site visit and looked at that application. We had talked about it but did nothing. We really didn't get to kick the tires on any examples; that was probably a mistake. If we had done that we would have at least raised some potential flags. We would have seen it in operation and talked to the end users that are using it and say "how do you really feel" about the new system. Is it slow, is it clunky? That would have given us a better comparison to what we had and what we could expect from this new system. That would start to get expensive especially if you have 2-3 finalists. The limited research and information on this aspect of project execution indicates that it is reasonable to spend anywhere from 3-5% of a total project budget on pre-project due diligence. $30-$50 thousand dollars on a million dollar project on trips and kicking the tires on a few systems may seem high but given the 200% schedule overage by vendor, and system performance concerns that would have been a good investment.

What had been talked about is if it got down to final 2-3 potential vendors that we should send both a technical and business representative to the sites and send the same people to all multiple sites so could come back and compare results to educate us. In order for that to occur, you need to be comfortable that you are not under time pressure to get started building a replacement system. After waiting for the number of years that MPD had, there was not a lot of tolerance for more additional delays. That could have cost us

dearly. There was pressure in the fact that every day, the old system was underperforming from a functionality perspective. In reality, we placed more pressure on ourselves then the old system did business operations-wise. When we originally scheduled the RFP we wanted to make sure that we weren't in the middle of the warm season or the winter season when we did the project. It really didn't matter as we had to navigate those trouble areas due to the Vendor's implementation problems anyway. We found a proposal that in many respects from our perspective helped to minimize the business impacts and more rapidly solve our angst of having to wait these number of years too. It was seductive by its very nature but had no basis in reality given the complexity of what we were trying to do.

Disciplined project estimation. Adequate performance by MPD — weak in some parts and strong in others.

We should have known that everyone else was a million dollars more and over a year in implementation. We took the vendor at their word. It looked like a good deal at the time. Inherent fallacy with how business is being done today. Derek wrote an article called "Can I have this Dance?" The article talks about the complicated relationship between the supplier and the customer. In order for the supplier to keep working and keep people employed, vendors have to respond to every reasonable opportunity to dance with the potential customer. The customer sets the project expectations and the vendor have to either accept those or not dance (cease to do business). Both parties come to the dance and after jointly setting unrealistic expectations that everybody has to live up to. Having done that, both parties sign a contract only to find themselves in bad situations later since neither party can delivery on those expectations without issues.

There is probably sin on both sides, but there certainly is sin when somebody says yes I can definitely do it for that price and this is how I'm going to accomplish it. Not sure how we protect ourselves from not being able to believe vendors when they say that they can do something. It's a troublesome situation. The only way you can protect one's interest is with knowledge and professional skills backed by best practices. But that is a tough school that costs time, money, and commitment. Even then, it is no guarantee since bad stuff happens on even the best projects. Costly experts that you bring to help one make good decisions can still make you look foolish and buyers that often don't meet their own commitments either are just as much to blame.

In MPD's case, Siebel can be implemented a hundred different ways. When Brian uses Oracle he does it the way that he knows how. When you bring someone in that knows how to do things just their way that's the way they're going to do it. Just like the project Vendor technical folks did. Other Vendor team members later added to the project would have done it entirely differently we later found out. Initially, MPD felt adequate to deal with the technical approaches proposed, but we learned in the end that we were weaker then we would have liked since we didn't know enough to question decisions when we should have.

START UP

Project plan including budget, schedule, change control, and risks. Overall adequate.

We did have a project plan which really was a schedule. The Vendor confused the two. At an early point in the project the schedule task progress ceased to be updated. MPD's Project Manager exerted pressure to get it updated, but development was under such a fast turn (another Agile development trap) that there were so many activities going that the vendor said, "My bad" I'll update later. For 3-4 weeks prior to what turned out to be "the melt down (UAT)" this was the case. MPD's concern was communicated to the executive leadership of the Vendor. The MPD Project Manager struggled to manage the project execution without an up-to-date schedule reflecting closely-held Vendor activities. The project was moving so quickly that by the time a schedule update would come out it was supposedly already out of date. It was this very weak adherence to discipline by the Vendor that tipped us in advance that something was wrong. The Vendor did not do a good job there.

We did have an overall project plan (different then a schedule) which was in reality was an enhanced Project Charter that was fairly elaborate and good document. We did a fairly good job on identifying risk and tracked it weekly.

At an informal communication level, MPD's Project Manager and IV&V were letting Dan Chelini know that although they couldn't prove it, they suspected that things weren't going well and to expect a slip. Spent ½ day in UAT testing before a project halt was called and was 9 months later before we entered UAT again.

MPD does think that we did do a good job with change control. In large part because MPD was willing to fight hard the first time the Vendor tried to nickel and dime us. MPD was also helped in reducing change because we did an adequate enough job with the requirements. It could have been a lot worse especially with a vendor that at that time was feeling the squeeze of profitability and was looking for any opportunity to get healthy or at least minimize the bleeding on their poorly estimated endeavor. *Note: The Vendor later intimated privately that they had given up what they felt were around $400K worth of potential project changes based upon MPD's initial defense of the first Change Order. This Vendor-provided value could not be validated so its validity is questionable.*

We could have done better and certainly should have held the vendor accountable ($ penalties) for accurate schedules that were updated and reflected reality. We even tried a couple of things utilizing IV&V by asking for quality metrics and conducting independent QA analysis but the vendor was very good at limiting our under the hood look into the project. Next time contract verbiage to allow unfettered access to project activities will be specified. The Vendor was formally saying everything was going well. We soon found out that they really didn't have the quality program that they said they did in their proposal. No real metrics. Their proposal said that they had a good quality program and they even had a person identified that was suppose to review things. In reality it wasn't there. Insight into developmental testing would have validated our suspicions earlier.

As mentioned earlier, the schedule discipline was weak. We didn't control that, the Implementing Vendor did. We tried to bring them into alignment repeatedly and did stay on them as far as risks and issues were concerned. There was a documentation trail with all the spreadsheets we were sending around including the bugs that helped the process as well as MPO for project artifacts.

Staffing / Training. Implementing Vendor was weak.

Training—within the first month MPD was at Siebel training as the Vendor suggested. Another double-edged sword because it allowed us to better understand Siebel; however, with the added length of the implementation (200% over original schedule) we had forgotten almost all the stuff we had training on. MPD virtually needed to relearn everything because it had been over a year since we went to training. The vendor said we needed to go so we did. MPD was at training within a month of the start of the project. Overall, we did an excellent job but it wasn't easy. MPD was always ready and didn't allow them to point the finger at us for any delays.

The vendor underestimated the work. MPD expressed concerns about the training. MPD had a difficult time getting good user documentation. The training program and documentation was questionable for a long time. The training was being developed at the same time that we were making changes to the test scripts by the same person. We had asked beforehand to have a specified person just for training. That person never really showed up.

When you see the Vendor's lead technician working 100 hours a week with a chest infection that later hospitalized him you know they're in trouble. They would be here early in the morning and stay until after midnight. Vendor staff health and demonstrable long hours were the informal metrics we used to determine trouble that was otherwise programmatically screened from us. That was a prime indicator to us that that the thing was not properly estimated or staffed.

When the same person that was responsible for training was also responsible for putting together all the technical documentation something is going to suffer. What gave was the training documents; scheduling; and all other project documentation.

Quality of documentation was the first thing to go. It was a time and money loser for them that they could get away with in every part of the project. If we didn't force them to deliver good documentation; they weren't going to. As soon as it becomes evident that the Vendor is not in a profitable situation, they start coping by cutting back on some things. Unfortunately for them on the other side we had a wonderful QA person in Anna Marie and they couldn't get by that. And from a contracts perspective we had that covered, they knew it so they had to deliver.

Where they also started cutting back was the testing. Where that became evident was at the first User Acceptance Test. MPD suspects that in all reality that was the reason that

we got into some of the design problems that we experienced. They were under such pressure to deliver that some things were not done smartly. Get it done as fast as possible. In retrospect the fastest way for them to get it done was to cause a lot of work on MPD's end. No elegance applied. That is why a lot of the processes were manual and took 30 steps to do some things instead of scripting those steps since time was wasting. They didn't have the cycle time and bodies so they took the path of least resistance.

This isn't a rocket science; it's a lot of art. We worked with the best information that we had at the time and in that context we acted to keep moving the project forward. It takes little intelligence or courage to be the Monday morning quarterback. Project work is a delicate balancing act. Does one move with courage or run away in fear when things get dicey?

Dependencies fulfilled for project start. Strong performance by MPD.

Based upon the information that we were given MPD was ready. The vendor side was very weak. The vendor didn't always give us the pertinent information especially concerning the hardware configurations.

MPD was strong. Dan and Brian had every server for Siebel in Development up and running when the vendor arrived.

Adequacy of the equipment and software. Weak performance by MPD and Vendor.

We had a statement about that early on related to the specs. Although we bought according to the specs, in retrospect we probably would have ordered differently knowing what we know today. The issue is that we probably did not get the right/appropriate information from the vendor and didn't question them enough. Basically, we thought we were going with the recommended solution not the minimum requirements. Don't confuse minimum with recommended. The Siebel software may have been adequate, and the problem may be the implementation but it's too expensive to figure that out completely.

We certainly did not have all the equipment requirements identified; otherwise we would not have had to buy all the servers we bought such as the Linux boxes. There are 21 servers for the total project, and they estimated 4 or 5 besides the ones for the Siebel portion of the project. We always knew we needed 8 servers for the Siebel implementation, due to a Development Environment, Test Environment and Production Environment.

For the telecommunications project the vendor originally estimated 1 server; we bought 2 so we could set up a mirrored environment. We were later told that 3 are needed. We bought 1 more because we had another other one for redundancy. But in terms of the servers over in the call center it did start out as saying you need 1, we bought 2. We actually addressed poor estimates at that time. We are above that amount over there now because of it.

The architecture for how the telecommunications system runs is different from the system that was already in place. It required a new design of the architecture so more hardware systems had to be bought to support the new requirements. This new architecture involved new domain and web controllers so a couple more servers were involved. That ate away at our project contingency funds.

In addition to the servers there were problems with configurations. They would say we needed something more. We would respond and ask them if a specific solution would work for them; if it was what they wanted. After they had responded yes, we went ahead with it. Literally at the go-live time we were finally told it won't work.

In retrospect we were weak. We did what we were asked to do, but there was a lot of engineering by the seat of their pants. There was not necessarily a lot of forethought. We trusted them and paid a steep price for it.

All of which goes back to the original staffing issue of not having the proper expertise on hand by the Vendors.

Review of the previous lessons learned. Strong performance by MPD.

MPD did not go through a formal Lessons Learned review process (it would help if the State had a repository for such things), but we did use the information learned through each iteration of the process, and the RFP was written based upon what was learned from previous successful projects. We did what we thought we could do but implementing a CRM's is difficult.

Given what MPD had available we did the best we could. We did try to cover ourselves by involving members of ITSD in the selection committee to approve the CRM; however, even the members of ITSD didn't know the CRM space well. MPD thought it had done a good job of reviewing the information that was available; however, there should have been more information available to us. And understanding the CRM space was one of the questions in the oral presentations that we asked all the vendors, what have you learned. We should have kicked the tires somewhere harder before we ever do something like this again.

Client camaraderie and flexibility. With the Implementing vendor, MPD was strong almost to the detriment that it looked weak in retrospect.

The technical team was very easy to get along with, and we could respect what we perceived to be really hard work on their end; however, activity does not necessarily equate to progress. And there was certainly no lack of activity which we thought was progress.

If you went to the end user side (MPD's Call Center Service Vendor), as far as camaraderie and flexibility, they were awesome. The call center was always willing to

do whatever they needed to do to make it happen. There were 2 end users at the call center that we had started with during the 3rd RFP and then the 4th RFP. They were really involved with it. In addition there are end users on MPDs side that were involved.

There was always a strong relationship between MPD and the end user. We tried to keep everyone in the loop as much as possible and sent out weekly communication pieces out to prepare the organization. Our users felt that we were competent and had good integrity with what we tried to do. We met the end user expectations as far as the project effort is concerned.

The mailroom and the front desk were the least informed in a lot of those cases. From an end users standpoint, we really focused on MPD's Call Center Service Vendor--more so than our internal users.

VENDORS

Relationships. With the vendors, it was strong.

MPD was blinded to our detriment because we gave them the benefit of the doubt. We were told that we needed to change our business rules. The business should not be a slave to the technology. We were constantly told the system does not work that way. Although we were open for a vanilla solution to changing business processes to make them line up if they made sense, we said we'd change it, but then we still ended up with tons of scripting and customization anyway.

Support. From a vendor perspective support was weak.

Otherwise Dan Bethke would not have spent the amount of time that he did trying to educate the people he did on web chat. MPD would not have been doing all that other support and buying all that equipment nor would MPD have been doing all the web development in many respects if we didn't have to.

Reliability. The vendor was weak.

The vendor lied to us. We felt it was adequate to strong until things went off course during the first UAT testing.

We knew that they were burning the midnight oil and thought that they were hard chargers, had a lot of respect for them. However, they didn't tell us a lot; they hid it. Although they shared a lot, it was all smoke and mirrors. Which is why both the MPD PM and IV&V went to Dan Chelini a couple of times to report that even though the vendor isn't saying so but we suspect to see a big change.

People lost their jobs over this project. This was one of those projects that had its own Boothill.

OTHER DEVELOPMENT TEAMS.

Level of Cooperation. Let's talk about the Vendors subcontractors in here. It was weak; poor execution.

It wasn't coordinated. God knows how many phone calls that we had to make and emails to various parties and how many times we had the same conversation with the telecommunications subcontract vendor and their support services. We were sick and tired of the poor patchwork type duct tape design that had gone into it that we demanded that the proper expertise be brought in to get it done. Only one subcontractor out of all of them did what he came to do and that was John T. And he is the only one until later as the project got over and then Tim showed up. Tim and John both did an excellent job based upon how weak it was to begin with.

If you look at to where we are today, it is still poor because we still don't know if we have support or not because we don't have a contact. MPD is operating without any contractual support. Which is what we had identified a risk a hell of a long time ago. In my opinion it is still weak for level of competence and cooperation because everyone is finger pointing at everyone else.

Our level of cooperation with them is good. We devoted 100% of our time.

MANAGEMENT AND CONTROL.

Procedures for orientation of new team members. Weak.

We didn't have any procedures and we had a lot of new members. It was trial by fire.

Procedures for evaluating task status and completion. Weak.

We could not get any insight from the vendor really where tasks were at. It was purely if we're done, we're not done or we're working on it. Nothing else was shared because they held that stuff really close. And when we asked and used IV&V to ask for evidence of that, we found out that it really wasn't there. It was real weak. On this particular point, their task evaluation was not supported.

Structure of Committees. Strong.

We did a pretty good job of structuring the committees. It worked. It was a good structure.

Frequency of status/staff meetings. Adequate.

The frequency of the meetings was very strong. We had a lot. It started out with a lot and then it died off. There was a period of time when it just went flat, for a month or two where they just disappeared. When they did disappear was when they realized that what

they reported to us so off schedule that they needed to hunker down to get the work done after we put the nail in it at UAT. We're not on; go fix your bugs.

It just leveled out to allow for more work and more meetings would not have corrected any problems. The main thing that would have helped any problem would have been more honesty at the meetings we had.

The communication with the project sponsors was probably more in the area of adequate. We did a good job of keeping them informed; we were very strong with the sponsor meetings in the beginning but we kind of trailed off. We were supposed to have them once a month. We did for awhile then we just flat-lined because we had nothing to report. The vendor had hunkered down; they were fixing stuff. The communication was very open there. It helped to have Anna Marie on both the steering committee and also a representative on the executive committee so she could take information from those meetings too.

Attendance at meetings. Strong overall from both sides.

We usually had people all dialed in. We even had it documented when people wouldn't have been there in the status report.

Procedure for meeting notices and preparation. Adequate.

We always needed to push the vendor's PM so we could distribute it. There were times that we were getting it while we were conferenced into the call, and they were copying it while we were all sitting around the conference table.

Effective meetings. Adequate.

The status meetings were fairly effective. The design meetings were worthless. They told us a lot without telling us anything. During the design meetings we would give our story and feel pretty good about things. Two weeks later they would tell us that we agreed to things that we never did as if we had signed off on things or bought into it. When it came time to demonstrate we didn't hold their feet to the fire. It was like, oh that piece is not working now, but it will and here's what it will look like. They had a lot of excuses and we gave them a lot of rope because we liked the guys. They ended up hanging us instead of them.

We would be hard pressed to find another team that put up with more or absorbed more hits and maintained the relations that our team did. If we would have given in to what we really wanted to do which was choke them a lot of times we would have given them the opportunity to walk. Or at least they would felt justified and taken it.

In talking with the technical team, they didn't care about the business part of it. They left the business part of it to the business people. They didn't care as long as they could get what they needed to get done.

Our relationship was just fine. That was successful. In many respects because of that we pushed them further than they ever wanted to go and when they tried to push back they ran into some difficulty. We made them work for some stuff. It wasn't one of those things that when they came back with a change or a different interpretation you didn't just swallow it. We made them work for every inch that they gained. Because of the relationship, they probably went further than where they should have gone from their management's perspective. It worked more to our benefit because we got more system for the buck. We could discuss performance all day long, but we got functionality.

Team attendance at end user meetings. Strong.

There was never an issue of people not supporting the meetings.

Staff input into decision-making and planning. MPD staff was strong.

The Vendor drove the schedule and MPD was always there to support them.

Timely and accurate project status reports. Weak

Vendor did not deliver on that aspect.

Status reports copied to team. Strong.

Everyone got them.

Isolation of team from management related problems. Strong.

Our technical team was isolated from potential problematic or programmatic areas from management areas. The technical guys could keep going while management argued over change control.

There were only a couple of times that MPD was told to stop work when push came to shove when we weren't supporting them. There were lots of conversations that took place that MPD technical team members were never aware of where the gloves would come off with the Vendor.

Timely checkpoints. Weak.

In retrospect we would have constructed the whole configuration review differently and used it as a quality gate for something else to go forward. We used test as the hammer, and it would have been nicer to have had a better opportunity in a design review. Although we did have the design review, we didn't have the proper mechanism to stop it at that point. It should not be "we'll just take your word for it, that this is not the way that it is going to be, we're just showing you the screens, the functionality is not part of this demonstration."

With hindsight being 20/20, we wouldn't have gone live when we did. Because of the data problems, we should have had that information in place and done prior to go live. We were at a breaking point with the vendor. It goes back to the earlier point of telling the truth, we were told that we had all the year one data in place on go live. Come to find out in December we still weren't done with year one data yet. Yet in September we were told that it was there. And that year 2 and 3 would be done by the following weekend yet is wasn't completed until months later.

There were multiple instances/opportunities for improvement.

Recruitment of new members. Not Applicable.

Management's sense of involvement. Strong.

Management was involved. The MPD Division Administrator and Commerce IT Manager were always there when we needed them. The Acting CIO wrote a letter. Wherever we had to tap into that power, it was there. From the Vendor side, when it went off the track in testing we had their attention with the vendor's Managing Director. Otherwise we wouldn't have had Scott and Russ. We had their attention. We didn't have telecommunications subcontract vendor's attention and still don't. The State didn't have any leverage with them.

Standards and procedures for general tasks. Not Applicable outside of our control.

It was the implementers. We didn't define the work plan; our job was to respond to their requests.

Responsiveness, striving for excellence. Weak.

We were always striving; however, excellence may not have always been the bar. A lot of it was just getting the thing to move. They met the letter of the requirement, but they didn't care how they got there.

From the vendor's management standpoint, they were more concerned about meeting the letter of the contract than what was being delivered as a product. They were not striving for excellence, otherwise we wouldn't have the Invite a Friend functionality that we have today. The technicians tried initially then fell off.

The vendor needed to temper it with the fact that they underestimated it so grossly that they were looking for exit strategies. At some point that they needed to draw a line. I don't think that I would be satisfied with my system being nicknamed Slowbel.

If I was striving for excellence I wouldn't be satisfied with a 34 step activity to do Invite a Friend. Striving for excellence doesn't end up with what we are going through with lead import or what we are putting up with printing labels right now. They were striving

for excellence right up to the point where they signed the contract or even up to the point of discovery.

At some point there was a moment on their end where they realized that they were going to lose money and that is when things turned around. Our requirements were well crafted and they hadn't thoroughly gone through them and when they finally had to implement them they realized boy we didn't take this into consideration and this is going to take a lot more work. That goes back to the sales team and the technical side too. As soon as profitability was an issue, excellence went out the door.

STAFF

Clarity of reporting relationships. Strong.

It was well documented in project charter from the beginning.

Communication among staff members. Strong.

Communication between staff and management. Strong.

Corrie did a great job of pointing out issues by being the devil's advocate which was value-added input. He did an excellent job of communicating the staff's interest at heart and the impact on MPD.

Communication between staff and end user. Excellent.

Recognition for a job well-done. Adequate

The MPD staff showed tremendous resilience. Not sweating the small stuff. All staff found a way to hang in there. It's a complicated system. When we get all trained up and tweak the system to get it where we need it we may end up loving system.

There are times when you have to let team and momentum take over. MPD stepped up. A lot of teams with potentially even more experience would not have seen it through the way you did. It was a class act for all of MPD.

We were recognized us in Monday staff meetings. The Administrator still wants to do a couple of things such as a press release and recognition event.

The team should be nominated for the Governor's Award not because we're good, but because we sat in a meeting with the CIO and were told that "CRM cannot be done properly in the State of Montana." It may not be a 10 project, but a successful one and if we get the remaining outstanding issues taken care of it's easily a 7. It was a successful CRM and first in the state.

Staff's use of consultants/contractors. Very strong.

Used Project Manager and IV&V. IV&V could only be there as a neutral third party and review documentation. Although we were hesitant using them in the beginning it was time and money well spent.

Morale. MPD was strong; the vendor's morale was weak.

MPD was more or less positive throughout. Overall MPD morale remained high because we didn't have a time constraint. Everyone thought that it was their role to do what they could to support Vendor and remain positive. But ongoing issues are causing some concerns with end users.

To the end users there was a certain amount of hesitancy from all of them since it was new. MPD's morale for the fulfillment and front desk has been poor since this has been so messed up on the fulfillment side. Some of the aspects system are that previously we had 300-400 packets going to Helena Industries for example and now we have packets of 15-30-50. Now we have to do things internally. We may say things to each other that we don't like, but when talking to the mailroom or the call center we tried to keep the morale up.

From the call center point of view, they're always optimistic. Call center agents have been positive, but they've also had their challenges with IVR and chat. It put a lot more on the call counselors because with all the problems other things have been let go.

Vendor's morale fell to very low point. Much of that was just pure mental and physical exhaustion; they were working 16-18 hour days all night long. Kristi was amazing she could work those hours, flying home every weekend. Sanjay and Jason were working all night long. They worked very hard.

Responsiveness, striving for excellence. Strong from MPD; not so for the vendor.

We responded to everything that we were asked to do very quickly. All members of the team rose to the occasion in their particular area that they were asked to do whether it was reading documentation, doing support on Siebel or Telecommunications, working with Rogie on the web stuff, whether it was testing. Everybody rose to where they needed to and everybody did the best that they could.

From the staff point of view we were really were responsive and strove to make this the best we could especially when it looked like we'd have to make lemonade out of lemons. I think that we had a vision of what we wanted in the end, and we always striving to reach that vision. At a point when we heard that "Siebel doesn't work that way" for the 500th time, we realized that we were going to have to compromise on some things.

On the vendors' side, again striving for excellence was not their main mission, meeting the requirement in the minimum way that they possibly could to save money was or so it appeared.

FACILITIES

Space for demonstrations, exhibits, and activities. Strong.

We didn't hear any complaints from the Vendor. We provided nice working space for 5 workers, equipment, and phones. A lot of places that they go they are stuck in a place where they are out in the open and in cubicles. In one place they went to, they didn't even have cubicles; it was just a bunch of desks sitting around each other. Good work space.

We did rent the computer training center the first time testing which was ideal. Better than having the space in the building because it took us off site so we could focus solely on testing. We didn't get back to that level of focus in testing later on. We were all totally psyched the first time. Afterwards we had lost so much steam and momentum by then due to the vendor missteps.

Space and ambience for staff See above.

Storage facilities. Not applicable.

Image facility communicates to clients. Not applicable

Length of lease agreement. Not applicable. Not specific to our project.

BUSINESS DESIGN

Early resolution of big issues. Weak.

Once we were made aware of issues we acted upon them quickly. However, we weren't told that we had big issues. They didn't crop up until late, until testing and demonstrations but we were told not to worry about it, because it would not look like that in the end. One issue that we had going into design was how we were going to integrate the database. There was resolution which proved to be the right thing to do which also ended up saving the vendor a ton of time and money.

Did vendor understand client's business requirements. Weak

Clearly not. MPD knew what we were talking about, but the vendor did not understand what we were talking about. In hindsight during the design meetings, we don't think that Kristi always understood the process we were trying to do. Otherwise the system would not have come out quite like it did. She took our requirements from the RFP and created the design documents without getting a lot of input from us. We could never see it to give her input. When she did show it to us, it was like this is the way it is going to be done as opposed to this is what our proposal is or is this going to work for you.

There was not enough early client review. We did have a business requirements document, but it was pretty skimpy. It had diagrams as far as the processes, and had a few requirements that met the specs. It met the letter, but not the spirit. When they presented it to us it was this is what you get whether you like it or not because Siebel doesn't work that way as opposed to will this work for you. There was a little of that, but it was pretty much this is the way that it has to happen in Siebel. This is the only way that Siebel can handle it.

Requirements engineering / Analysis. Weak.

The engineering side met all the requirements minimally, but they did not engineer them to be user friendly. They did it the easiest, cheapest way they could. They apparently didn't do very good analysis on them or we didn't give them the information that they needed, back to the performance and complexity problems. If we would have given them that, then the requirements would have been met to more what we thought they should have been. They didn't strive for excellence.

Access to legacy/batch systems. Strong.

They had full access to the legacy system until cutover and even after. It was limited because they had to figure out the architecture of the legacy system; Allie had to go through and map out the entire database. They had full access to the legacy system as well as Oracle and guestbook in Access. We didn't even talk about how much MPD assisted in converting our online entry points to match Siebel.

That is part of the web stuff that we did in house that they didn't have to do. It was above and beyond the database integration that we did. We did it because it was the only way that Siebel could work. Initially the onus of that was on the vendor and MPD took it on. We did 100% of the front end and then Rick told Rogie how to program the backside to allow it into Siebel.

Formal sign-off. Strong.

Everything was constructed to have a formal signoff. We didn't hold their feet to the fire, but we didn't fully understand what we were signing off on. Yes the information that they had there was correct in as far as it was concerned, but if there would have been more information and more details in the business design document then it would have had been control all the way through. We didn't realize that anything that was not in the business design document was not going to be implemented until we had already signed off on it. It was implied that this represented the design, but wasn't exactly the design because they could never show us exactly what the design was.

They still can't show us the exact design of some things. If you look at the design document now, it does not look like what we have in place in almost any of the system.

TECHNOLOGY / METHODOLOGY

Leading edge methods and procedures. Did they follow best practices and in project management in terms of process. – Strong.

The quality of the documents could have been better.

Having Derek as a project manager and Mike as IV&V was nice and very strong. It was nice to have someone involved who wasn't part of MPD so they could look at things with an external and impartial view. As well as someone who had been through a similar process before who could guide us in what we needed to be looking for. In spite of the problems with the project, the methodology and procedure was good.

There was room for improvement had we known more we could have done things differently. That was a shortcoming of the person on the vendor's side that was responsible for communicating. The first project manager from the vendor's side did not communicate with us or with his supervisors. We're not sure about how he was with the internal team, but they had to know what was going on because they were working on it.

Compliance with industry standards. Adequate.

The vendor was supposed to have quality assurance people on their side, but that didn't happen. The reason that first project manager got into trouble was that he wasn't keeping either side informed of where the project was and all the sudden it blew up on him.

Meeting the technical challenges. Weak.

Failed. Very weak due to the vendor's misunderstanding of our requirements. It goes back to the conversation on staffing. They couldn't meet all of the technical challenges because they didn't have enough technical positions on site or working on the project to meet them in an appropriate fashion.

Toolset. Adequate.

Siebel is strong in certain instances, but for what we needed it wasn't the right fit. It didn't meet our expectations. The toolset is much more complicated and cumbersome than envisioned or portrayed. That also goes to the sales switch and also to the training that was required. How many weeks have we had to train the front desk, call center, mailroom? They really didn't meet that aspect of it.

Toolset is referring to the application itself that the end user uses. It is adequate here because the toolset in Siebel like the server configuration and server management is not hard to use. Sometimes it's hard to find things because you don't know where it is sitting, but in general if the programming toolset is adequate then the Technology/Methodology is very adequate. The application itself is way easier to use than tools.

The application is cumbersome and complicated due to the amount of scripting.

PROGRAMMING

Standards and code reviews. Very weak.

MPD never saw a piece of coding. It was suppose to be vanilla but is 70% and 80% scripted.

SCM/Build/Release process. Weak.

SCM is a type of version control. Did they show us versions? We didn't see any of that until we did iterative UAT testing. It was discussed during status meetings, but we never saw it.

Suitability of language to end-product. Not applicable because it is was the only choice with Siebel.

MPD can edit the scripts, but not anything internal to Siebel. MDP will be going to e-scripting classes.

Expected performance understood and methods for proving achievement. Weak.

Weak because it wasn't lined out specifically in the specs so they didn't have to meet any. There was a general performance requirement to the whole system but not specific for individual components.

Toolset (coding, unit testing, integration, SCM, etc.) and environment. Toolset refers to Siebel tools and the scripting; the tools to develop it. -- Adequate.

For the stuff that is vanilla the system tools that you can look at to see how things are running. But for the list import as soon as you hit the promote many button you have to go in to look at the log files to figure out anything because it is not vanilla.

Unit testing, integration testing. Weak.

In unit testing they fix something but because of all the scripting it breaks something else. Unit testing was poor and complicated. Before we went to UAT they didn't test anything.

They didn't obviously do any before they had us originally start UAT. Their original time was too short to allow for adequate testing.

USER DOCUMENTATION

Clear, concise. Strong

It is clear and concise now, but required lots of revisions. The documentation that we have is pretty good. The end user guides have step by step instructions with screen shots. The administration guide is good too, all the processes in it don't work, but the guide itself is good.

The telecommunications subcontract vendor documentation is scattered, but it can be done. They originally didn't any plan to do at all. When asked during the original telecommunications subcontract vendor install if they were documenting anything in Missoula, we just received a look.

SYSTEM TEST

Test Plan. Weak on the Vendor's part.

There was a test plan, but they were not prepared to go into testing.

The test plan was drawn up in one of the status meetings. The plan was there, but it wasn't followed prior to UAT. The MPD PM and IV&V kept asking where is the plan?

Accounting for stress testing and performance achievement Weak.

Never did that. After production data loads we found out that it was having problems. We had clues on that based upon data transition that we were going to have problems especially with live loads.

Criteria for test completion. Strong.

We had good criteria, but it took a lot of revisions to get there.

DEPLOYMENT

Installation/Acceptance test. Weak.

Both scheduled go live times, the vendor wasn't ready. The first time was because they weren't finished developing the application. The second time was because the data wasn't loaded. To our credit the installation was done well before cutover. During final acceptance testing, it was too short a time period and we had our back against the wall. We capitulated by allowing them to do some really big bug fixes after go live. Not a good ideal. We still have bugs.

Training. Adequate

Kristi and MPD were all there for deployment training. It went well. The system was easy enough to use that it didn't require that much training. Brad and Lana had also been using the system during testing. End user training was strong.

The administration training was weak. Rushed, we really didn't get into the list import because not working, template fields being updated. We sensed that the trainer had more important things to do than admin training, but he did do it.

Cut-over. Weak.

We had all the data issues. We should not have cut over on the date that we did cut over. Telephony was down the morning that we went live. Brian had to troubleshoot that morning. That was the date that we now realize that we have some problems that still exist. When we stopped the cut-over the first time it was because of the data import problem. We still had data problems the day before. We went live thinking that had one year of data, in December it still was not done.